

Motion-tracked Speed Dialing for Smartphones

Soong-Hee Lee¹, You-Ze Cho²

¹Dept. of Information and Communications Engineering, Inje University
Gimhae city, Gyeongnam, 621-749 Korea
[e-mail: icshlee@inje.ac.kr]

²School of Electronics Engineering, Kyungpook National University
Daegu, 702-701 Korea
[e-mail: yzcho@ee.knu.ac.kr]

Received July 2, 2014; revised August 6, 2014; accepted August 26 2014; published October 31, 2014

Abstract

The use of a smartphone with one hand has been getting difficult as the size of the smartphone display gets larger. In addition, people handicapped or using one hand for handling other pieces may encounter difficulties to touch exact numbers for calling with their smartphones. Thus, motion-tracked speed dialing can be a good solution for these cases and also an advanced feature applicable in the future network not requiring dedicated Internet connections. The previous study on this feature showed a result with a limitation of using servers connected via Internet connections for solving real-time control issues. Therefore, this paper proposes a scheme for the smartphone dialing on motion-tracked numbers without any servers or Internet connections, and presents the details of its implementation and results.

Keywords: Smartphone, Motion-tracked speed dialing, Motion tracking

This work was supported by the Inje Research and Scholarship Foundation in 2014. We express thanks to Mr. Yoon-Ho Nam, Mr. Soo-Yeop Kang, Mr. Han-Seong Lee, Ms. Song-Hwa Park who assisted the implementation.

<http://dx.doi.org/10.3837/tiis.2014.10.013>

1. Introduction

With the trend of expanding size of the display device, the use of smartphones with one hand has been getting harder than before. In addition, people handicapped or using one hand for handling other pieces may encounter difficulties to touch exact numbers for calling with their smartphones. Thus, motion-tracked speed dialing can be a good solution for these cases and appears as a future network application for smartphones. Voice recognition features can also be applied for speed dialing, that have already become quite advanced as shown in the case of Siri. Notwithstanding, motion-tracked speed dialing can still be very useful in terms of offering customized services for handicapped users with background noise or for emergency calls that require silence.

The previous study in this issue, number recognition using accelerometer of smartphone[1], directly uses accelerometer values of each motion for matching with the model patterns. A pre-processing scheme is proposed to reduce erroneous cases in matching that requires heavy computations in this study. This burden leads to the use of a server connected with the calling smartphone via Internet connections that ultimately limits the free use of this feature especially in the future network after IP[2].

Motion tracking through accelerometer in a smartphone was also explored in recent studies [3]~[6], mainly applied at the large scale applications such as urban street behaviors, movement pattern recognition, or map tracking. The driving style recognition using a smartphone as a sensor platform was also explored in [7]. Moreover, it has been proposed to use a smartphone as a 3D input device and the concept of 3D navigation is examined in [8]. However, it has been hard to find the applications that recognize these motions for a basic feature such as calling to the abbreviated destination number except [1].

On the viewpoint of number recognition, there have been many researches on the recognition of hand-written or typed numbers[9]~[15]. Technologies such as fuzzy logic[9] or neural networks[10][11][13] have been applied to recognize hand-written[9][10] or typed numbers[11]~[15]. These studies have been mainly assumed to be used for specific purposes such as recognition of license plate numbers[11][14][15]. Even though these studies have shown good results on the recognition rate, there have hardly been any research on the recognition of the numbers constructed from user's motion.

The authors have already explored the possibility of using the accelerometer for motion tracking in smartphone games[16]. The basic theory for tracking the motion of smartphones in this study is expected to be effectively applied into this case of motion-tracked speed dialing to minimize the processing burden that prevents the daily use of the motion-tracked speed dialing in the environments without Internet connections.

This paper proposes a scheme for the motion-tracked speed dialing that requires no Internet connections by storing the accelerometer values in each measuring instance and tracking the motion of the smartphone device, i.e., the x-z coordinates, calculated from the stored sensor values, and comparing with each pattern of numbers. The next chapter talks about the basic principles used for the proposed scheme for motion-tracked speed dialing with mathematical presentations. Then the implementation methods of the proposed scheme are explained. The implemented result and the conclusion are lastly given.

2. Basic Principles for the Motion Tracking

For the successful implementation of the intended scheme, the basic principle must be clarified first. This paper mainly focuses on the three aspects of the proposed scheme, i.e., motion tracking from sensor values, storing before motion tracking and decision making, for the purpose.

First, it is required to consider the scheme to track the motion of the smartphone device in real time. The coordinates of the current position of the smartphone device could be considered for the motion tracking[16]. The current position of the device can be expressed as the values of x, y, z in the 3-dimensional space. The initial position needs to be fixed to the starting point of the x, y, z coordinate system for reference.

Let the coordinate values of initial position and the initial speed be $(0, 0, 0)$ and 0 , then the center of the device is given as $s_0 + v_0 t + \frac{1}{2} a t^2$ where the acceleration, speed and distance are given as a, v_0, s_0 . At $t=0$, the distance on the x -axis from the zero point and the speed of the device are assumed to be $x_0 = 0, u_0 = 0$. We just focus on the x -axis here for starting.

After $t = \Delta t$ elapsed, the distance from the zero point and the speed of the device are given as $x_1 = \frac{1}{2} a_1 (\Delta t)^2, u_1 = a_1 \Delta t$ where the acceleration is measured as a_1 on the x -axis.

After $t = 2\Delta t$ elapsed, the distance from the zero point and the speed of the device are given as $x_2 = x_1 + u_1 \Delta t + \frac{1}{2} a_2 (\Delta t)^2, u_2 = u_1 + a_2 \Delta t$ where the acceleration is measured as a_2 on the x -axis.

After $t = 3\Delta t$ elapsed, the distance from the zero point and the speed of the device are given as $x_3 = x_2 + u_2 \Delta t + \frac{1}{2} a_3 (\Delta t)^2, u_3 = u_2 + a_3 \Delta t$ where the acceleration is measured as a_3 on the x -axis.

Therefore, after $t = k\Delta t$ elapsed, the distance from the zero point is given as

$$x_k = x_{k-1} + u_{k-1} \Delta t + \frac{1}{2} a_k (\Delta t)^2 \quad (1)$$

where the speed and distance from the zero point at the next previous instant are given as u_{k-1}, x_{k-1} and the acceleration is measured as a_k on the x -axis.

Similarly, after $t = k\Delta t$ elapsed, the distance from the zero point on the y -axis is given as

$$y_k = y_{k-1} + v_{k-1} \Delta t + \frac{1}{2} b_k (\Delta t)^2 \quad (2)$$

where the speed and distance from the zero point at the next previous instant are given as v_{k-1}, y_{k-1} and the acceleration is measured as b_k on the y -axis.

The distance from the zero point on the z -axis is also given as

$$z_k = z_{k-1} + w_{k-1} \Delta t + \frac{1}{2} c_k (\Delta t)^2 \quad (3)$$

where the speed and distance from the zero point at the next previous instant are given as w_{k-1} , z_{k-1} and the acceleration is measured as c_k on the z-axis (c_k must be measured excluding the acceleration of gravity). The location of the smartphone device, after $t = k\Delta t$ elapsed, is decided by the coordinates (x_k, y_k, z_k) , indicating the location of its center.

Second, the burden for processing to track the position must be reduced to avoid the Internet communications with a server which is required especially in the future network environments without IP connections[2]. The most portion of processing burden is used for pre-processing to match with the model patterns in the previous study of [1], which leads to the use of Internet communications with a server to support the dialing smartphones that ultimately limits the free use of this feature especially in the environments without Internet connections. Hence, it is needed to reduce such burdens to incorporate all necessary functions into a smartphone device that enables the proposed scheme to be applied as a future network application for smartphones.

3. Considerations for Practical Applications

Smartphone devices include various capabilities such as sensors and communications. The proposed scheme in this paper utilizes acceleration sensor for motion tracking and communications for establishing connections to intended users via the smartphone devices.

Considering the possible directions of motions shown in Fig. 2, one of three axis needs to be excluded for tracking the motion of the device since only two axis values are required to specify one of the numbers to be dialed. For conveniences in implementation, we choose x and z axis values for specifying dial numbers while the y axis value is excluded.

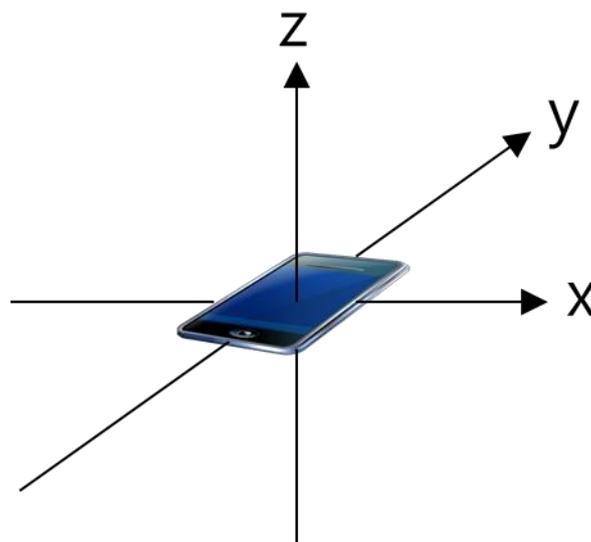


Fig. 2. Possible moving directions of smartphone devices in the 3-dimensional space.

The motion tracked results from the equations (1) and (3) after trying a number “2” are shown in the Fig. 3. The results show such a variety that seems to be hard to be easily decided

as the number “2”. This leads to the difficulties in trying pattern matching methods widely used in general number or character recognition systems[9]~[15]. Therefore, another way of approach needs to be explored for the proposed scheme.

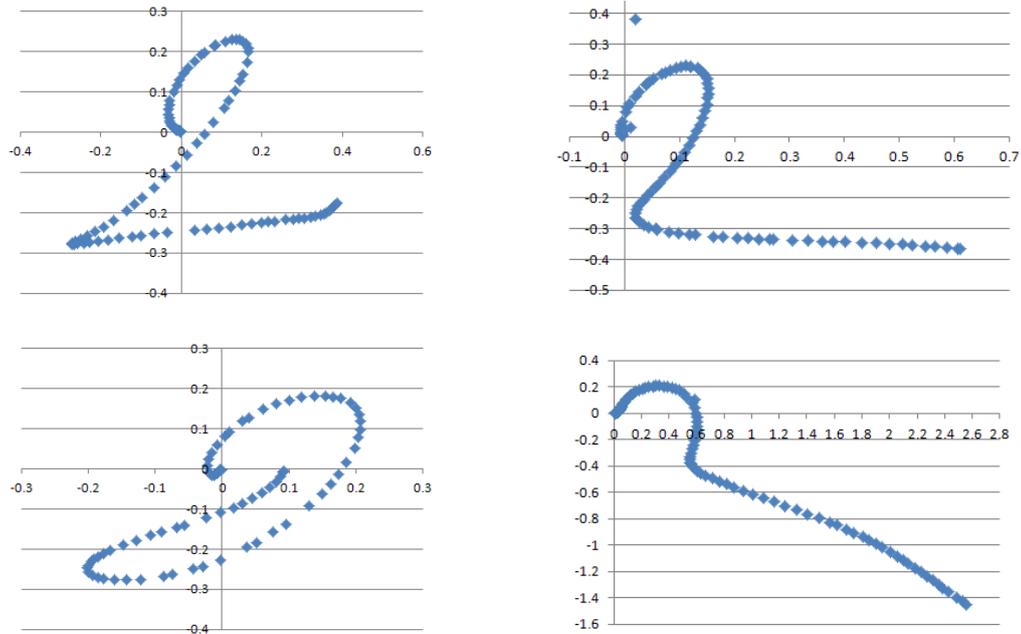


Fig. 3. Various patterns of motion tracked results for a number “2”.

4. Implementation of the Recognition Scheme for Motion-tracked Speed Dialing

The previous sections identified the basic principle for the proposed recognition scheme and considerations required for practical applications. In addition to these, the implementation according to the proposed scheme demands more detailed methods for recognizing numbers for dialing in motion-tracked numbers.

For actual implementation, the acceleration sensor in the smartphone device needs to be considered. The acceleration sensor produces sensed values for each axis. Then the location and speed of the device can be calculated from the acceleration data as explained in the section 2. To reduce pre-processing burdens which leads to the use of Internet communications with a server, model patterns of numbers for matching, as shown in the **Fig. 4**, are proposed for simplicity and more efficient implementation.

Generally, it is inappropriate to use Java, the programming language used for application developments in android-based smartphones, for real-time processing. Hence, the fastidious considerations are required for reading sensor data and tracing the motion of the smartphone device in real time. The previous study[1] utilizes a server connected with the calling smartphone via the Internet connection to avoid real-time calculations inside the device that ultimately limits the free use when not connected to Internet. In contrast, the proposed scheme in this paper has no limits on the use even when not connected to Internet as it temporarily stores the sensor data into the memory buffer to avoid the processing such as calculations that

make real-time processing difficult while the screen is touched and stayed. The stored data is to be calculated and processed to decide the location and direction on each coordinate after the touch is detached from the screen.

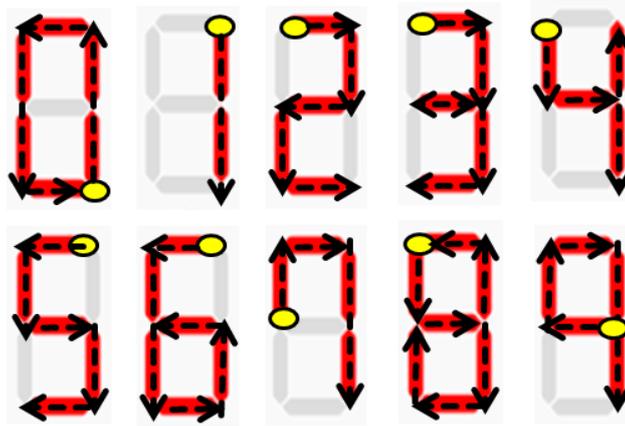


Fig. 4. Proposed model patterns of numbers for matching.

Applying all of these considerations, the recognition scheme for dialing in motion-tracked numbers is designed and the overall procedure of the designed scheme is as shown in the Fig. 5.

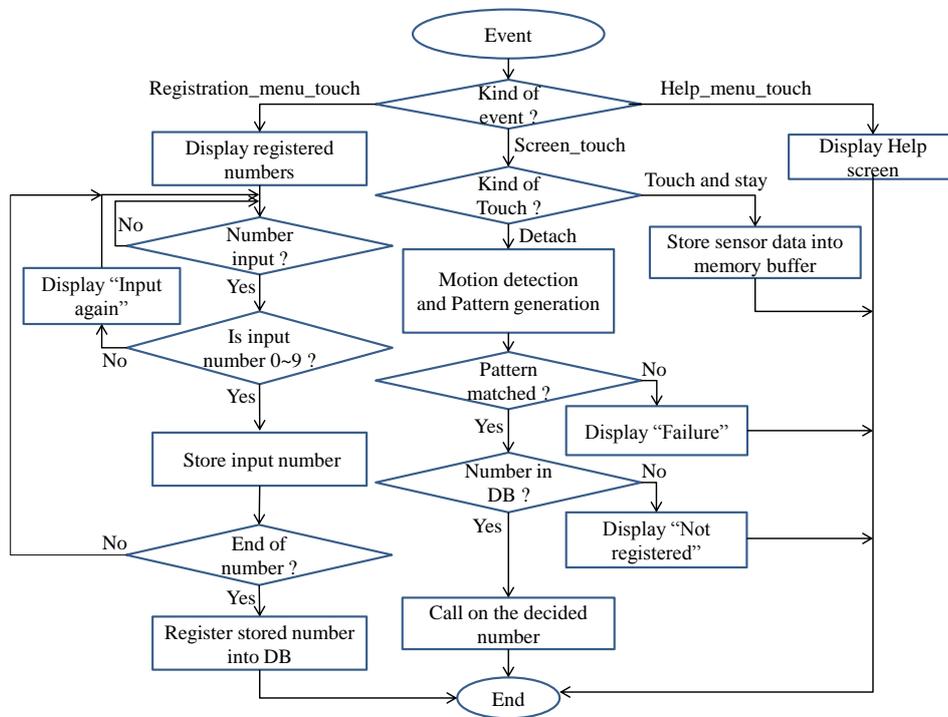


Fig. 5. Overall procedure of the proposed scheme.

The proposed scheme is run based on several events, including screen touch and menu touches. When one of these events occur after the initial screen, it is first classified into one of three cases, i.e., help_menu_touch, registration_menu_touch, and screen_touch. In case of help_menu_touch, help screen is to be displayed for users trying this application first. When registration_menu_touch occurred, the screen showing registered phone numbers is displayed and the phone number input by the user is registered as one of the speed numbers. There are two types in the screen_touch event including “Touch and stay” and “Detach.” The “Touch and stay” action causes all of the data from acceleration sensor to be sequentially stored into the memory buffer to be used in the following detection procedure.

The “Detach” action triggers the main functions of the proposed scheme, i.e., motion detection, pattern generation/matching and calling. The procedure for the motion detection and pattern generation can be expressed as the following algorithm:

Input:

- 1) The stored sensor values.
- 2) The coordinates of the current device location (initialized to 0).
- 3) The current direction of the device (initialized).
- 4) The indexes of each direction (initialized to 0).

Output:

- 1) The pattern for the matching.

Procedure:

- 1: while (uncalculated sensor data left in the memory buffer)
- 2: Calculate the current location coordinates of the device by applying eq. (1) and (3).
- 3: if the current location is less than the previous one on the x-axis
- 4: The Left_index is incremented.
- 5: The Right_index is cleared to 0.
- 6: The Up_index is cleared to 0.
- 7: The Down_index is cleared to 0.
- 8: elseif the current location is larger than the previous one on the x-axis
- 9: The Right_index is incremented.
- 10: The Left_index is cleared to 0.
- 11: The Up_index is cleared to 0.
- 12: The Down_index is cleared to 0.
- 13: else if the current location is less than the previous one on the z-axis
- 14: The Down_index is incremented.
- 15: The Right_index is cleared to 0.
- 16: The Left_index is cleared to 0.
- 17: The Up_index is cleared to 0.
- 18: elseif the current location is larger than the previous one on the z-axis
- 19: The Up_index is incremented.
- 20: The Left_index is cleared to 0.
- 21: The Right_index is cleared to 0.
- 22: The Down_index is cleared to 0.
- 23: endif
- 24: if the Right_index is larger than α
- 25: The current direction is set to “Right” and added to the pattern.
- 26: elseif the Left_index is larger than α
- 27: The current direction is set to “Left” and added to the pattern.

```

28:   elseif the Up_index is larger than  $\alpha$ 
29:     The current direction is set to "Up" and added to the pattern.
30:   elseif the Down_index is larger than  $\alpha$ 
31:     The current direction is set to "Down" and added to the pattern.
32:   endif
33: endwhile

```

The output of the presented algorithm, the pattern for the matching, is a kind of string that includes characters indicating detected directions. The direction is decided only after the value of index for each direction exceeds a certain value α (set to 3 in the implementation) that prevents possible misrecognitions due to the transient fluctuations of the sensor data.

After the motion detection and pattern generation, the generated pattern is to be matched with the model patterns shown in the Fig. 4. To implement this matching procedure, basic patterns consisting of directions such as Right, Left, Up, Down, as shown in the Table. 1 are prepared. On attempts to test the implemented software, there have been found some deviant patterns generated from hand's tiny vibrations or unintended habits that cause misrecognition problems. Hence, more patterns are empirically added to lower miscognition rates as shown in the Table. 1 after numerous attempts.

Table 1. Matching patterns for each number.

| Number | Basic patterns | Added patterns |
|--------|-----------------------|---|
| 0 | Up-Left | - |
| 1 | Down | - |
| 2 | Right-Down-Left-Down | Right-Down-Right-Down |
| 3 | Right-Down-Left-Right | Down-Left-Right |
| 4 | Down-Right-Up | Down-Up Down-Up-Down Down-Right-Down-Up |
| 5 | Left-Down-Right | Left-Down-Down Left-Right-Down-Down |
| 6 | Left-Down-Right-Up | Left-Right-Down-Right-Up-Right Left-Down-Up Left-Right-Down-Down-Up Left-Down-Down-Up Left-Down-Right-Down-Up |
| 7 | Up-Right-Down | - |
| 8 | Down-Right-Down | Down-Down |
| 9 | Left-Up | Left-Right-Up-Right-Down |

When a match is found between the output pattern and the matching patterns of the Table. 1, calling to the registered number is started. If not found, the "Failure" message is displayed on the screen. Fig. 6 shows the images of implementation results. After the application is installed and run, the initial screen of (a) is displayed. For the users who want to know the way of using the application, the selection button on the left shows awake the help menu screen of (b). The selection button on the right invokes the registration menu screen of (c). On any screen of (b) and (c), the motion menu on the left-top side bring to the initial screen of (a). Then user can touch and stay on the screen except the selection menus for drawing the intended number in motion until the end of the drawing. The image of (d) shows the in-calling screen when the registered number for the detected speed number is 000-0000-0000.

To confirm the usability of the proposed scheme, it is required to verify the usefulness through checking the ratio of correct dialing. First, the test is performed after showing only the help screen to the users. The resulted ratios are shown in the dark bars for the general users in the Fig. 7. Next, the test is performed after repeating the motions for each character for training. The results are shown in the light bars for the trained users. The x-axis shows the reduced numbers, 0 to 9, while the y-axis shows the ratio of correct dialing in the Fig. 7. These results were obtained after the tests had been repeated 200 times for each case. The results show some variations depending on the number. Numbers with simple or unique patterns such as 1, 4, 7, 9 show better results while numbers with the similar patterns such as 2, 3 and 5, 6 show lower recognition rate. This implies that the more delicate decision rule is required to obtain the better results on the numbers showing lower recognition rates. The averages of the results for trained users stay around 87% implying that the proposed scheme may apply to the general uses in daily life without Internet connections.

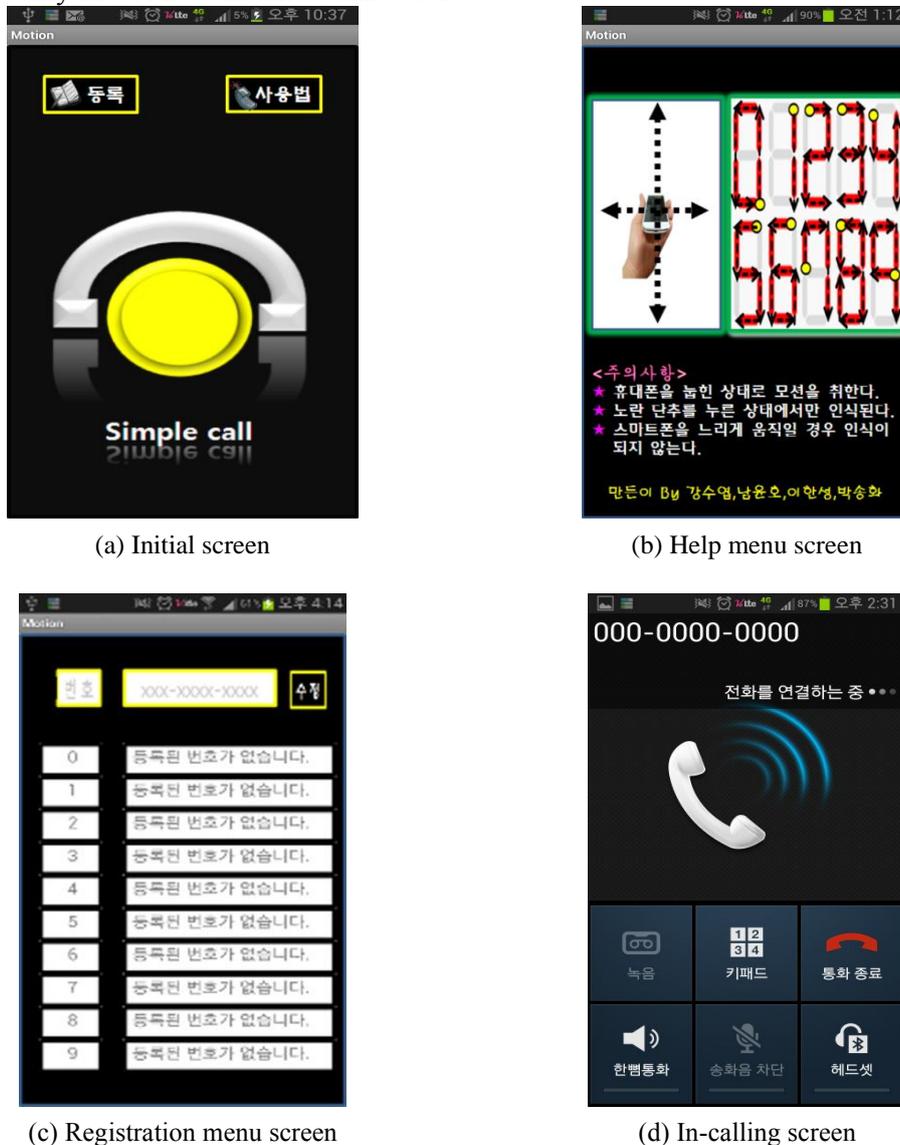


Fig. 6. Screen images of the implementation result.

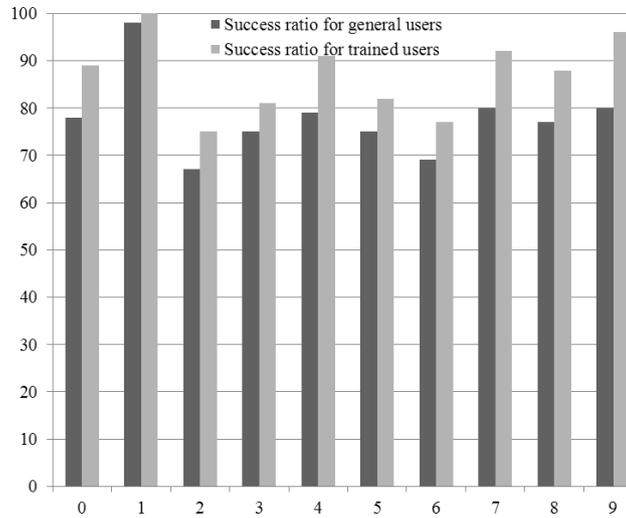


Fig.7. Results of the calling tests.

5. Conclusion

A new scheme for motion-tracked speed dialing for smartphones is proposed and ways for implementation including algorithm and considerations for practical applications are explored in this paper. In contrast with the previous study, the proposed scheme in this paper has no limits on the use even when not connected to Internet as it temporarily stores the sensor data into the memory buffer to avoid the processing burdens such as pre-processing calculations that make real-time processing difficult while the screen is touched and stayed. In addition, the proposed scheme tried to minimize the processing burden by simplifying the motion recognition mechanism through only decision of direction and model matching. We tried to verify the usefulness of the proposed scheme by testing for both of ordinary users and trained users. The test results show the average ratio of correct dialing of 87.1% for trained users that may be used for general uses in daily life though it is not better than the result of the previous study. It seems that there exists a trade-off between recognition rate and the processing burden. The more elaborate mechanisms for deciding the numbers are expected to be studied for the better ratio of correct dialing overcoming this trade-off barrier.

References

- [1] Seok Chan Bae, Bo-Gyung Kang, "Number Recognition Using Accelerometer of Smartphone," *Journal of the Korean association of information education*, vol. 15, no. 1, pp. 147-154, Feb., 2011. [Article \(CrossRef Link\)](#).
- [2] Ji Zhaojun, "“After IP” Innovation and ICA based Future Network," *China Communications*, Feb. 2005. [Article \(CrossRef Link\)](#).
- [3] Armir Bujari, Bogdan Licar, Claudio E. Palazzi, "Movement Pattern Recognition through Smartphone's Accelerometer," in *Proc. of Consumer Communications and Networking Conference (CCNC), 2012 IEEE*, pp.502-506, January, 2012. [Article \(CrossRef Link\)](#).
- [4] Hyekyung Yang, Hwanseung Yong, "Physical Activity Recognition using Accelerometer of Smart Phone," in *Proc. of Korea Computer Congress 2012*, vol. 39, no. 2(D), pp. 7-9, June, 2012. [Article \(CrossRef Link\)](#).

- [5] Jun Han, Emmanuel Owusu, Le T. Nguyen, Adrian Perrig, Joy Zhang, "ACComplice: Location Inference using Accelerometers on Smartphones," in *Proc. of Communication Systems and Networks (COMSNETS)*, pp.1-9, January, 2012. [Article \(CrossRef Link\)](#).
- [6] Armir Bujari, Bogdan Licar, Claudio E. Palazzi, "Road Crossing Recognition through Smartphone's Accelerometer," in *Proc. of Wireless Days (WD), 2011 IFIP*, pp.1-3, October, 2011. [Article \(CrossRef Link\)](#).
- [7] Derick A. Johnson and Mohan M. Trivedi, "Driving Style Recognition Using a Smartphone as a Sensor Platform," in *Proc. of 2011 14th International IEEE Conference on Intelligent Transportation Systems*, pp.1609-1615, October, 2011. [Article \(CrossRef Link\)](#).
- [8] Henning Graf, Prof. Dr. Klaus Jung, "The Smartphone as a 3D Input Device," in *Proc. of IEEE Second International Conference on Consumer Electronics*, pp.254-257, September, 2012. Article (CrossRef Link). [Article \(CrossRef Link\)](#).
- [9] Paul Gader, James M. Keller, Juliet Cai, "A Fuzzy Logic System for the Detection and Recognition of Handwritten Street Numbers," *IEEE Transactions on Fuzzy Systems*, Vol. 3, No. 1, February, 1995. [Article \(CrossRef Link\)](#).
- [10] Jian Wang, Jingfeng Yang, Shaofa Li, Qiufang Dai, Jiaying Xie, "Number Image Recognition Based on Neural Network Ensemble," in *Proc. of Third International Conference on Natural Computation (ICNC 2007)*, 2007. [Article \(CrossRef Link\)](#).
- [11] Zheng-He Song, Bo Zhao, Zhong-Xiang Zhu, En-Rong Mao, "Research on Traffic Number Recognition Based on Neural Network and Invariant Moments," in *Proc. of Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*, Hong Kong, 19-22 August 2007. [Article \(CrossRef Link\)](#).
- [12] Yun Li, Mei Xie, "An Effective Method For Number And Letter Character Recognition," in *Proc. of 2008 11th IEE International Conference on Communication Technology Proceedings*, 2008. [Article \(CrossRef Link\)](#).
- [13] Ting-ting Zhao, Ji-yin Zhao*, Rui-rui Zheng, Lu-lu Zhang, "Study on RMB Number Recognition Based on Genetic Algorithm Artificial Neural Network," in *Proc. of 2010 3rd International Congress on Image and Signal Processing (CISP2010)*, 2010. [Article \(CrossRef Link\)](#).
- [14] Yi-Chong Zeng, Jing Fung Chen, "Adaptive Template-matching Method for Recognition of Low-resolution License Number," in *Proc. of 2010 International Computer Symposium (ICS)*, pp. 228- 233, Dec. 2010. [Article \(CrossRef Link\)](#).
- [15] Shoaib Rehman Soomro, Mohammad Arslan Javed, Fahad Ahmed Memon, "Vehicle Number Recognition System for Automatic Toll Tax Collection," in *Proc. of 2012 International Conference on Robotics and Artificial Intelligence (ICRAI)*, pp. 125-129, Oct. 2012. [Article \(CrossRef Link\)](#).
- [16] Soong-Hee Lee, "Combining Communications and Tracking: A New Paradigm of Smartphone Games," *KSII Transactions on Internet and Information*, Feb., 2013. [Article \(CrossRef Link\)](#).



Soong-Hee Lee received the B.S., M.S., and Ph.D. degrees from Kyungpook National University, Korea in 1987, 1990 and 1995, respectively. From 1987 to 1997, he was a member of research staff in Electronics and Telecommunications Research Institute. Since 1997, he has been with the Dep't of Information and Communications Engineering, Inje University. His research activities are in the area of next generation network technologies and services. He was designated as an IT Standard Expert on behalf of Korea by Ministry of Information and Communications in 2001. He is an Editor in the ITU-T SG13 and a Member of KSII.



You-Ze Cho received the B.S. degree in Electronics Engineering from Seoul National University, Korea, in 1982, and the M.S. and Ph.D. degrees in Electrical Engineering from the Korea Advanced Institute of Science and Technology, in 1983 and 1988, respectively. Since 1989, he has been with the Kyungpook National University, Korea, where he is currently a Professor of School of Electronics Engineering. From August 1992 to January 1994, he had been at the University of Toronto, Canada, as a visiting researcher. From March 2002 to February 2003, he had been at the National Institute of Standards and Technology (NIST), USA, as a guest researcher. His research interests include mobility management and traffic engineering for wireless and mobile networks, broadband convergence networks, future Internet, and wireless ad-hoc networks.